

IN THE CLAIMS:

1.(Original) A method for preventing net update oscillation of a bus bridge by competing net update messages, said method comprising the steps of:

(a) determining whether a particular portal is a coordinator on its local bus; and proceeding to step (b) (i) if said particular portal is a coordinator, otherwise, proceeding to step (b) (ii);

(b) (i) determining whether said particular portal finds a net update collision on its local bus; and proceeding to step (c) if the net update collision is found;

(b)(ii) determining whether the particular portal receives an UPDATE_ROUTE message from another portal that is a coordinator on the local bus; and proceeding to step (c) if the UPDATE_ROUTE message is received;

(c) setting a global net_update bit to one by a lock procedure;

(d) verifying whether the lock procedure in step (c) has been successfully performed by determining whether the net_update bit has been set to one;

(e) performing one of:

(i) discarding the net update if it has been determined in step (d) that the lock procedure in step (c) has not been successfully performed; and

(ii) processing the net update according to IEEE1394.1 bridge standard and setting the net_update bit to zero.

2. (Original) A method of preventing net update oscillation of a bus bridge by competing net update messages, comprising:

(a) receiving on a first portal of said bus bridge a first net update message from a first coordinator on a first bus;

(b) receiving on a second portal of said bus bridge a second net update message from a second coordinator on a second bus before said first net update message has been processed by said first portal of said bus bridge;

(c) selecting and processing by the bridge of one of the first and second net update messages as a surviving net update message, and discarding the other of the first and second net update messages; and

(d) updating clan information so that both the first and second portal contains clan information of the surviving net update message.

3. (Original) The method according to Claim 2, wherein step (c) includes initiating a reset of one of the first bus and second bus that had a discarded net update message.

4. (Original) The method according to Claim 3, further comprising:

(f) initiating a reset of the bus from which the portal of the survived net update message has been selected in step (c).

5. (Original) The method according to Claim 2, wherein the net update message being received first in time is selected in step (c).

6. (Original) The method according to Claim 2, wherein the bridge comprises one Central Processing Unit (CPU), and the net update message found first by the CPU is selected in step (c) as the surviving net update message.

7. (Original) The method according to Claim 2, wherein the bridge comprises a multiple Central Processing Unit (CPU) configuration, and the net update message that is first reported by one CPU to at least one other CPU in the multiple CPU configuration is selected in step (c) as the surviving net update message.

8. (Currently Amended) The method according to Claim 2, wherein a configuration of the first portal and second portal of the bridge designates one of the first portal and the second portal for selection of its net update message in step (c) to be designated as the surviving net update message.

9. (Original) The method according to Claim 2, wherein the net update message selected in step (c) as the surviving net update message is selected by prime selection criteria according to IEEE1394.1 draft standard, except that the bridge does not edit any update messages.

10.(New) An apparatus for preventing net update oscillation of a bus bridge by competing net update messages, the apparatus comprising:

a processor in communication with a memory, the processor executing code for:

setting a global net_update bit to one by a lock procedure when a net update collision is determined;

verifying whether the lock procedure has been successfully performed by determining whether the net_update bit has been set to one;

discarding the net update when that the lock procedure has not been successfully performed.

11. (New) The apparatus according to claim 10, wherein the processor further executing code for:

processing the net update according to IEEE1394.1 bridge standard and setting the net_update bit to zero.

12. (New) The apparatus as recited in claim 10, wherein the code is stored in the memory.